

THE UNIVERSITY OF MELBOURNE
SCHOOL OF COMPUTING AND INFORMATION SYSTEMS
COMP90043 CRYPTOGRAPHY AND SECURITY

Assignment 2, Semester 2 2021
Hints and Solutions

READ VERY CAREFULLY: Questions 1d and 1e will be *auto-marked*, ie, marked by software, *not* a person. Thus, the format of your answer is *strictly required* to be of the following form: you must download the template txt file from the canvas (from the same place as this Assignment is found), and then paste your integer solution for Question 1d in the second line, immediately after the text “Signature:” (with no space). Similarly, your solution to Question 1e must be written immediately after the text “Blind Signature:” on the third line, (again with no space). It is also *imperative* that you fill in your student number in the first line, immediately after the text “Student number:” (with no space). The code used to mark the answer will use this line containing the student number, so make sure that you have entered it correctly. Also, do not put any spaces *after* any of the integers you enter.

Also, this Assignment is split into 2 Sections, the *graded questions* and the *ungraded questions, exam style*. Only your solutions to the questions in the Graded Questions Section will be marked, and only these marks will contribute toward your grade for the class. The remaining questions are highly encouraged and are written in exam style. It is suggested that feedback for these questions are sought by your tutor, or by whoever is running a future consultation session.

Graded Questions

1. [18 marks] We discussed in the subject that a message M (encoded as an integer) can be signed using RSA private key by $S = M^d \bmod n$ and verified by the corresponding public key by $M' = S^e \bmod n$ and check whether $M == M'$. We also showed the concept of blinding on RSA.

Perform the following implementation tasks in a language of your choice. You are not allowed to use any library function to perform exponentiation. In order to get full marks, your algorithm has to be able to work in realistic cryptographic environments (consider inputs in the order of 10^{1000}).

- (a) Implement the function $\text{SIGN}(M, n, d)$ which takes the encoded message and a private key as arguments, calculates and returns the signature. You may assume that $M < n$. Print the code here.
- (b) Implement the function $\text{VERIFY}(S, n, e, M)$ takes as inputs a signature, a public key and the original message. It should return `TRUE` if the signature is valid, or `FALSE` if the signature is invalid. You may reuse the $\text{SIGN}()$ function implemented above, if you want to. Print the code for this function.

- (c) Implement a function `BLINDSIGN` to sign a message using the blinding concept. Remember, you shouldn't directly sign the original message by $M^d \bmod n$ as you did in `SIGN` function, but you may call the `SIGN` function if needed. Print the code here.

An integer encoded message M and a pair of RSA keys are provided as following:
 $M = 314159265????93$ (please replace `?????` with the last five digits of your student ID)

Private key: $\langle n, d \rangle$

Public key: $\langle n, e \rangle$

$n = 11396311342906819133245180752504625094447926145771153608337005942535340$
 $831151082124611648733795917345423093120647809492578196651328326613421541984$
 $374544599265256494866003364648970813971670451048426724934881335069848815008$
 57942197501

$e = 65537$

$d = 20729576806810227945651433503304642530313216592724403339332811669890870$
 $507980537712665435487675836653308618504240738644446969730044899317107941502$
 $247799584959444798172916891463972996495752944622965018659022099059225470003$
 8562058305

- (d) Sign the message by calling your above implemented `SIGN` function. Print the signature here.
- (e) Sign the message through blinding process by calling `BLINDSIGN`, with x equals to the last four digits of your student ID (you may find the definition of x from week 5 lecture). Print the signature for the blinded message.
2. [18 marks] Alice wants to both sign and encrypt a message M for B . Assume Alice's public key is K_A and Bob's public key is K_B and they have already shared public keys with each other correctly. She is considering two options:
- Approach A (Sign then Encrypt): Alice computes $S = \text{Sign}(K_A, M)$, then $C = \text{Encrypt}(K_B, M||S)$ and sends C to Bob.
- Approach B (Encrypt then Sign): Alice computes $C = \text{Encrypt}(K_B, M)$ and $S = \text{Sign}(K_A, C)$ and sends C, S to Bob.
- One of these two approaches doesn't successfully authenticate that the message M originally came from Alice. Show how an active attacker can modify what Alice sends to Bob to trick Bob into thinking the message came from the attacker.
3. [14 marks] Consider the following hash function based on RSA. The key $\langle n, e \rangle$ is known to the public. A message M is represented by blocks of predefined fixed size $M_1, M_2, M_3, \dots, M_m$ such that $M_i < n$. The hash is constructed by XOR the results of encrypting all blocks. For example, the hash value of a message consisting of three blocks is calculated by

$$H(M) = H(M_1, M_2, M_3) = (M_1^e \bmod n) \oplus (M_2^e \bmod n) \oplus (M_3^e \bmod n)$$

Does this hash function satisfy each of the following requirements? Justify your answers (with examples if necessary).

- (a) Variable input size
- (b) Fixed output size
- (c) Efficiency (easy to calculate)
- (d) Preimage resistant
- (e) Second preimage resistant
- (f) Collision resistant

Ungraded questions, exam style

1. Professor Parampalli generated two pairs of RSA keys for his tutors, using a pair of p and q . The chosen public component e_1 and e_2 are different prime numbers. Both p and q are very large prime numbers and were destroyed immediately after generating the following keys:

Will: $\langle n, e_1 \rangle, \langle n, d_1 \rangle$
Wenjun: $\langle n, e_2 \rangle, \langle n, d_2 \rangle$

Answer the following questions with detailed process and/or justification.

- (a) Professor Parampalli wants to send a confidential message M to both Will and Wenjun, so he calculates and then sends $C_1 = M^{e_1} \bmod n$ and $C_2 = M^{e_2} \bmod n$. Explain how you may recover this message without knowing Will's or Wenjun's private key.
 - (b) Outline a strategy that may help Will recover Wenjun's private key.
2. An alternative key distribution method suggested by a network vendor is illustrated in Figure 2.
 - (a) Describe the scheme in steps.
 - (b) How do A and B know that the key is freshly generated?
 - (c) How could A and B know that the key is not available to other users in the system?
 - (d) Does this scheme ensure the authenticity of both A and B? Justify your answer.
 3. Assume that Alice has chosen a large RSA modulus n such that factorization is impossible with reasonable time and resources. She also then chooses a large random public exponent $e < n$ for which the RSA problem is also not practical. However Bob decides to send a message to Alice by encrypting each alphabet character (represented by an integer between 0 and 25) separately using Alice's public key $\langle n, e \rangle$.

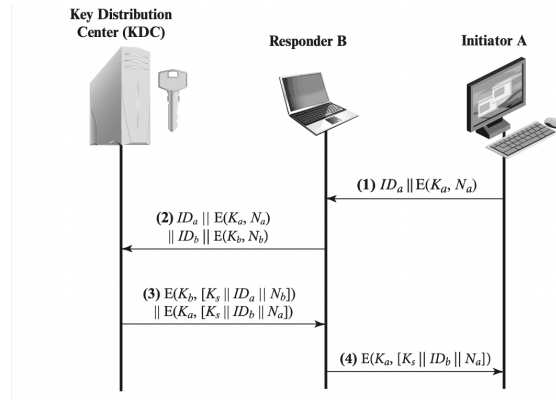


Figure 1: Scheme

- (a) Describe an efficient attack against this method.
- (b) Suggest a countermeasure to this attack.

Submission and Evaluation

- You must submit a PDF document via the COMP90043 Assignment 2 submission entry on the LMS by the due date. Handwritten, scanned images, and/or Microsoft Word submissions are not acceptable — if you use Word, create a PDF version for submission.
- Late submission will be possible, but a late submission will attract a penalty of 10% available marks per day (or part thereof). Requests for extensions on medical grounds will need to be supported by a medical certificate. Any request received less than 48 hours before the assessment date (or after the date) will generally not be accepted except in the most extreme circumstances.
- This assignment will be marked out of 50 marks, and will contribute to 7.5% of your total marks in this subject. Marks are primarily allocated for correctness of your thinking and clarity of your communication, rather than (only) the correct result without sufficient justification.
- We expect your work to be neat — parts of your submission that are difficult to read or decipher will be deemed incorrect. Make sure that you have enough time towards the end of the assignment to present your solutions carefully. Time you put in early will usually turn out to be more productive than a last-minute effort.
- You are reminded that your submission for this assignment is to be your own individual work. For many students, discussions with friends will form a natural part of the undertaking of the assignment work. However, it is still an individual task. You are welcome to discuss strategies to answer the questions, but not to share the work (even draft solutions) on social media or discussion board. It is University policy that cheating by students in any form is not permitted, and that work submitted for assessment purposes must be the independent work of the student concerned. Answers which are not original may not receive full marks even if they are correct.

Please see <https://academicintegrity.unimelb.edu.au>

If you have any questions, you are welcome to post them on the LMS discussion board *so long as you do not reveal details about your own solutions*. You may also email the Head Tutor, Will Troiani (william.a.troiani@gmail.com) or the Lecturer, Udaya Parampalli (udaya@unimelb.edu.au). In your message, make sure you include COMP90043 in the subject header. In the body of your message, include a precise description of the problem.